Amendments to the Claims

Please amend Claims 1, 8, 11, 18, and 20. The Claim Listing below will replace all prior versions of the claims in the application:

Claim Listing

1. (Currently Amended) A method of modeling a digital processor comprising:

loading operations and data representations of a target processor defined in a high level programming language, the operations and data representations being used in commands for execution by the target processor; and

in a computer system, using the defined operations and data representations, simulating the target processor executing certain commands to provide a model of the target processor data representations and operations to decrease development time of a real-time implementation of the certain commands on the target processor, the simulating including generating model data indicative of results of the target processor having executed the certain commands, the generated model data providing a bit level representation of the target processor results.

2. (Original) A method as claimed in Claim 1 wherein the step of simulating includes providing the model generated data in human readable terms instead of machine code.

3. (Previously Presented) A method as claimed in Claim 1 further comprising the step of executing working code on the target processor, such that the target processor generates working data, and

wherein the step of simulating includes generating model data corresponding to the working data generated by the target processor in a manner such that the model data is (i) bit-wise matchable to the target processor generated working data, and (ii) in human readable terms.

4. (Original) A method as claimed in Claim 1 wherein the step of simulating further includes using the high level programming language, defining data types for the data representations of the target processor.

5. (Original) A method as claimed in Claim 4 wherein the step of simulating further comprises the steps of:

for a given source processor, (a) determining each distinct fixed bit length data representation, and (b) grouping the determined distinct data representation to form a set;

for each target processor, repeating steps (a) and (b) such that respective sets are formed; and

forming a hierarchy of the formed sets by correlating one set to another such that a base class with depending subclasses are generated and form the hierarchy, each set being defined by one of the base class and a subclass.

6. (Original) A method as claimed in Claim 4 wherein the step of defining operations is incremental such that one target processor operation at a time is defined and modeled using the high level programming language.

7. (Previously Presented) A method as claimed in Claim 1 further comprising the step of generating diagnostic data corresponding to the simulating.

8. (Currently Amended) A method as claimed in Claim 7 wherein the step of generating diagnostic data includes indicating the number of times different operations of the target processor are encountered during the simulating.

9. (Original) A method as claimed in Claim 1 wherein the step of simulating is incremental, such that a first set of certain data representations and operations of the target processor is simulated using the high level programming language to form an intermediate model of the target processor, and subsequent to the formation of the intermediate model, at least a second set of data representations and operations of the target processor is simulated

using the high level programming language to increment the intermediate model toward a final desired model of the target processor.

10. (Cancelled)

11. (Currently Amended) A computer-readable medium having stored thereon sequences of instructions, the sequences of instructions including instructions that, when executed by a digital processor, cause the processor to perform:

loading operations and data representations of a target processor defined in a high level programming language, the operations and data representations being used in commands for execution by the target processor; and

in the processor, using the defined operations and data representations, simulating the target processor executing certain commands to provide a model of the target processor data representations and operations to decrease development time of a real-time implementation of the certain commands on the target processor, the simulating including generating model data indicative of results of the target processor having executed the certain commands, the generated model data providing a bit level representation of the target processor results.

12. (Previously Presented) The computer-readable medium according to Claim 11 wherein the instructions causing the processor to perform simulating the target processor include instructions that cause the processor to perform providing the model generated data in human readable terms instead of machine code.

13. (Previously Presented) The computer-readable medium according to Claim 11 wherein the instructions further include instructions causing the processor to perform executing working code on the target processor, such that the target processor generates working data, wherein the instructions causing the processor to perform simulating the target processor include generating model data corresponding to the working data generated by

the target processor in a manner such that the model data is (i) bit-wise matchable to the target processor generated working data, and (ii) in human readable terms.

14. (Previously Presented) The computer-readable medium according to Claim 11 wherein the instructions causing the processor to perform simulating the target processor further include instructions that cause the processor to perform, using the high level programming language, defining data types for the data representations of the target processor.

15. (Previously Presented) The computer-readable medium according to Claim 14 wherein the instructions causing the processor to perform simulating further include instructions that cause the processor to perform:

for a given source processor, (a) determining each distinct fixed bit length data representation, and (b) grouping the determined distinct data representation to form a set; for each target processor, repeating steps (a) and (b) such that respective sets are formed; and

forming a hierarchy of the formed sets by correlating one set to another such that a base class with depending subclasses are generated and form the hierarchy, each set being defined by one of the base class and a subclass.

16. (Previously Presented) The computer-readable medium according to Claim 14 wherein the instructions causing the processor to perform defining data types is incremental such that one target processor operation at a time is defined and modeled using the high level programming language.

17. (Previously Presented) The computer-readable medium according to Claim 11 wherein the instructions further include instructions that cause the processor to perform generating diagnostic data corresponding to the simulating the target processor.

18. (Currently Amended) The computer-readable medium according to Claim 17 wherein the instructions causing the processor to perform generating diagnostic data further include instructions that cause the processor to perform indicating the number of times different operations of the target processor are encountered during the simulating the target processor.

19. (Previously Presented) The computer-readable medium according to Claim 11 wherein the instructions causing the processor to perform simulating the target processor further include instructions that cause the processor to perform (i) simulating in an incremental manner, such that a first set of certain data representations and operations of the target processor is simulated using the high level programming language to form an intermediate model of the target processor, and (ii) subsequent to the formation of the intermediate model, simulating at least a second set of data representations and operations of the target processor using the high level programming language to increment the intermediate model toward a final desired model of the target processor.

20. (Currently Amended) An apparatus for modeling a digital processor, comprising:

a source of operations and data representations of a target processor defined in a high level programming language, the operations and data representations being used in commands for execution by the target processor; and

a computer system using the defined operations and data representations, the computer system simulating the target processor executing certain commands to provide a model of the target processor data representations and operations to decrease development time of a real-time implementation of the certain commands on the target processor, the computer system generating model data indicative of results of the target processor having executed the certain commands, the generated model data providing a bit level representation of the target processor results.

21. (Previously Presented) The apparatus as claimed in Claim 20 wherein the computer system provides the model generated data in human readable terms instead of machine code.

22. (Previously Presented) The apparatus as claimed in Claim 20 further including a target processor in communication with the computer system, wherein the target processor executes working code such that the target processor generates working data, and wherein the computer system generates model data corresponding to the working data generated by the target processor in a manner such that the model data is (i) bit-wise matchable to the target processor generated working data, and (ii) in human readable terms.

23. (Previously Presented) The apparatus as claimed in Claim 20 wherein the computer system uses the high level programming language and defines data types for the data representations of the target processor.

24. (Previously Presented) The apparatus as claimed in Claim 23 wherein:

for a given source processor, the computer system (a) determines each distinct fixed bit length data representation, and (b) groups the determined distinct data representation to form a set;

for each target processor, the computer system repeats steps (a) and (b) such that respective sets are formed; and

the computer system forms a hierarchy of the formed sets by correlating one set to another such that a base class with depending subclasses are generated and form the hierarchy, each set being defined by one of the base class and a subclass.

25. (Previously Presented) The apparatus as claimed in Claim 23 wherein the computer system defines operations incrementally such that one target processor operation at a time is defined and modeled using the high level programming language.

26. (Previously Presented) The apparatus as claimed in Claim 20 wherein the computer system generates diagnostic data corresponding to simulating of the target processor.

27. (Previously Presented) The apparatus as claimed in Claim 26 wherein the diagnostic data includes indications of a number of times different operations of the target processor are encountered during the simulation of the target processor by the computer system.

28. (Previously Presented) The apparatus as claimed in Claim 20 wherein the computer system simulates the target processor incrementally, such that the computer system simulates a first set of certain data representations and operations of the target processor using the high level programming language to form an intermediate model of the target processor, and subsequent to the formation of the intermediate model, the computer system simulates at least a second set of data representations and operations of the target processor using the high level programming language to increment the intermediate model toward a final desired model of the target processor.